Status of the pixel read-out code (Pomone)

Copyright © 2003 United Feature Syndicate, Inc.

G. Chiodini[1], B. Hall[2], S. Magni[3], D. Menasce[3], L. Uplegger[3], D. Zhang[2]

[1]I.N.F.N. Lecce, [2]FNAL, [3]I.N.F.N. Milano

What is Pomone anyway?

Vertumne et Pomone, by Anthony Van Dyck 1599-1641
Oil on canvas
Galleria di Palazzo Bianco, Genova

Pomone was the roman goddes of fruits and vegetables and was capable of transforming herself depending upon the surrounding environment, simbolizing adaptation and resilience to ever changing conditions.

Where we are

The system is now ready to be deployed to users as version 1.0

After a few stumbling blocks the system now works as per design.

- we had to align the Linux versions of the machine in Milano with the one at Fermilab.
- We upgraded to the latest version of the g++ compiler (3.2)
- We upgraded to a new version of the Xerces (XML) parser (a major revision)
- Functionality of the GUI has considerably been improved

- The system is able to take both real data at the MTEST, with several detectors at once, and also to perform complete or selective calibrations.

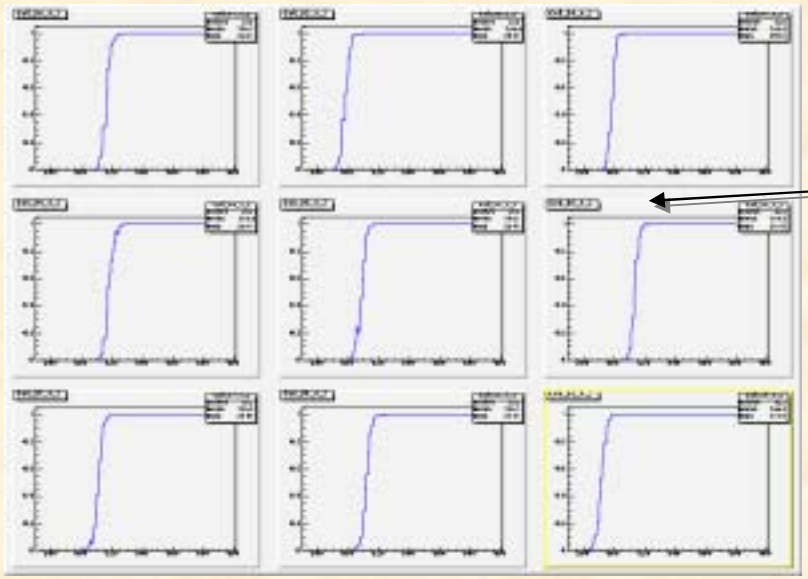
- Users interact with the system with completely functional GUI

An extensive Reference Guide has been provided to allow people to extend the functionality of the system and we are on our way to provide also a User Guide. (live demo at the end)

We used the system to perform a full fledged calibration of FPIX1 chips both at **Feynman** and the **MTEST**

The two test-stands are twins in all respect: the major difference is that at MTEST we are forced to use long cables and a feed-through board to drive and read-out the detector, while at Feynman we can decide to use short cables instead and avoid the feed-through board (this allows us to measure the amount of additional noise we get from those)

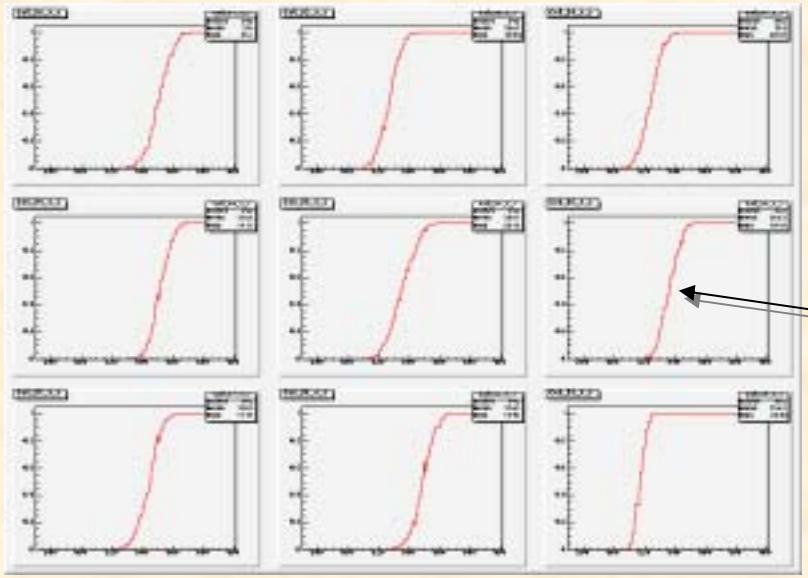
These tests already allowed us to make significant progress in debugging the read-out system (Brad recently fixed a problem diagnosed in the mezzanine FPGA code)



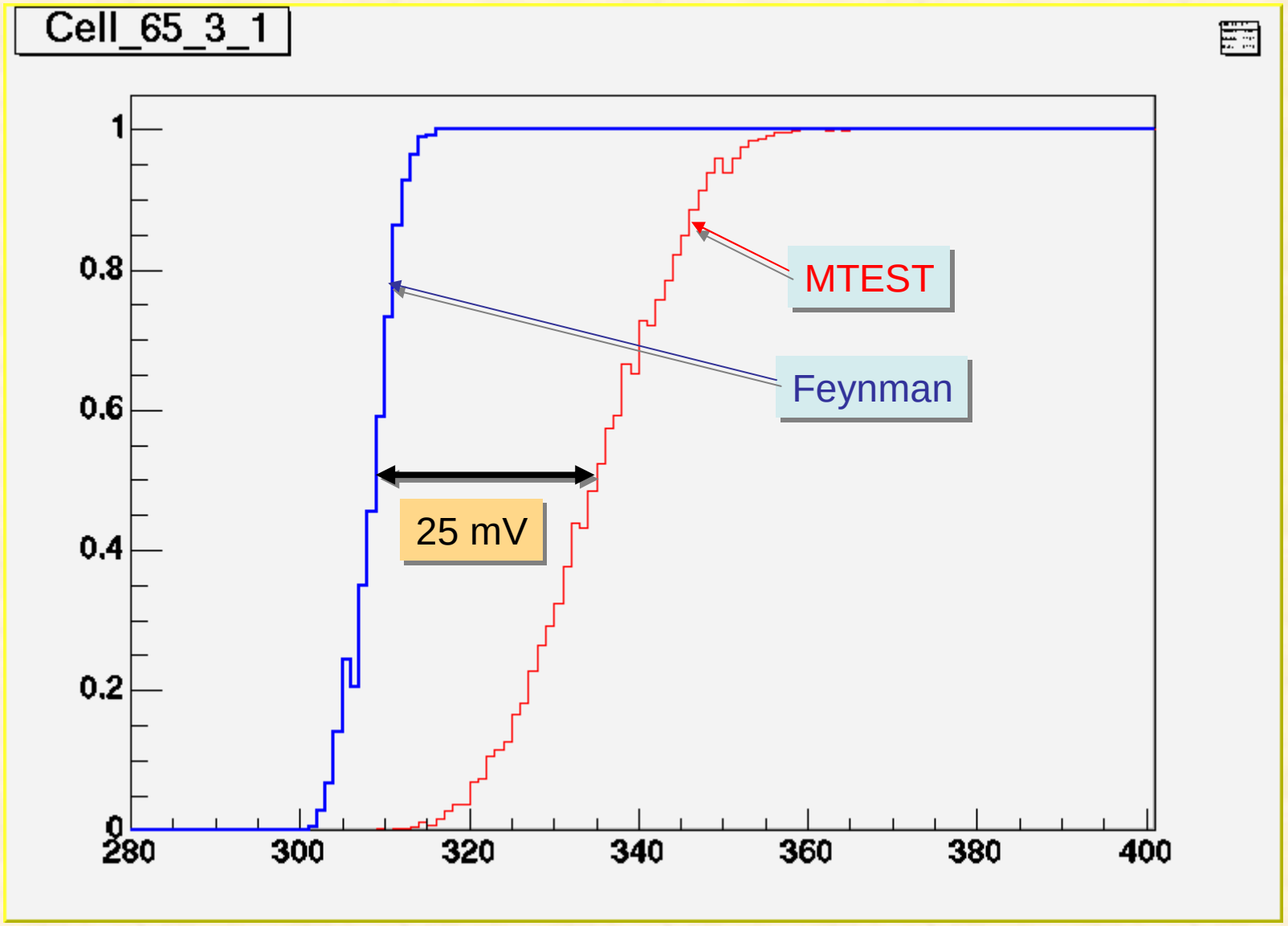
Measurements taken at Feynman with both long cables and feed-through board, but smaller number of Lemo connectors with respect to MTEST

A distinct difference is visible:

at MTEST the electronic chain introduces a significant amount of noise and the pulser signal is more attenuated



Measurements taken at MTEST with both long cables and feed-through board, in the final setup configuration



Yesterday's measurement: next step is systematic investigation of noise performances

The Pomone GUI

The GUI is ready to be deployed for general use (Brad is already routinely using it).

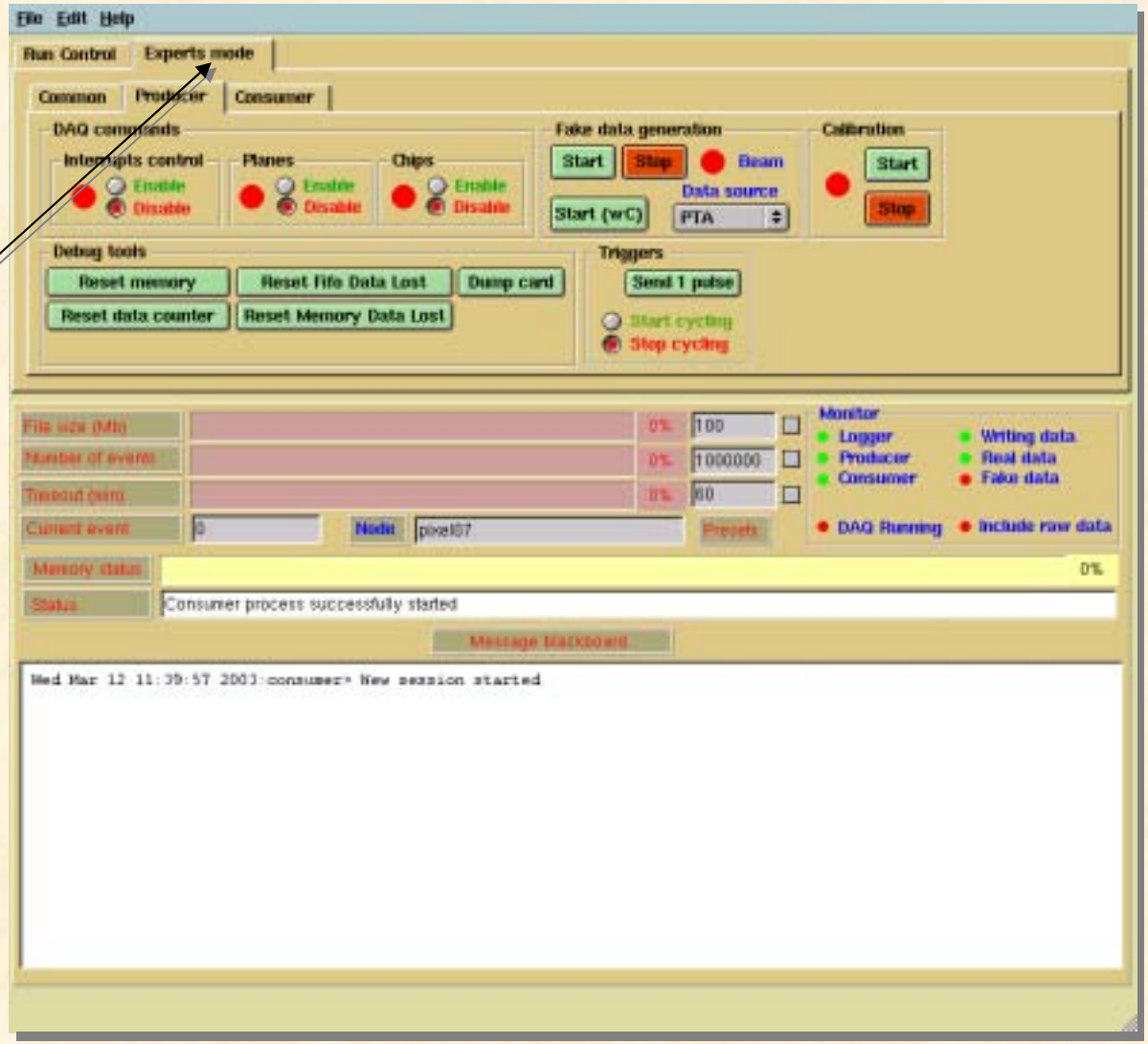
There is an average-user mode (run control) as well a more sophisticated experts mode.

The Pomone GUI

The GUI is ready to be deployed for general use (Brad is already routinely using it).

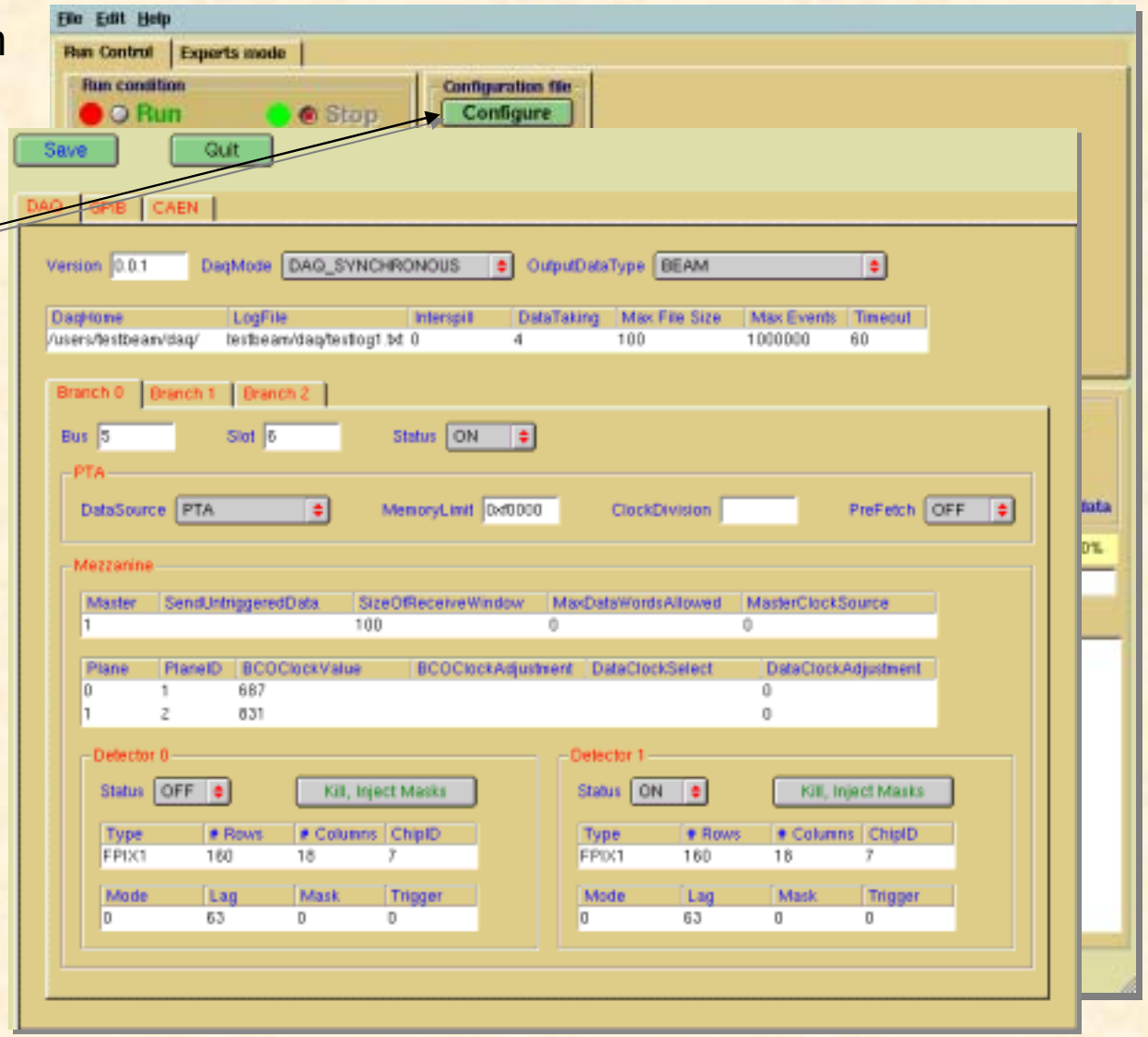
There is an average-user mode (run control) as well a more sophisticated experts mode.

The expert mode has additional tabs to control individual components of the DAQ, for fine-tuning, normal-mode override and debugging/troubleshooting



The Pomone GUI

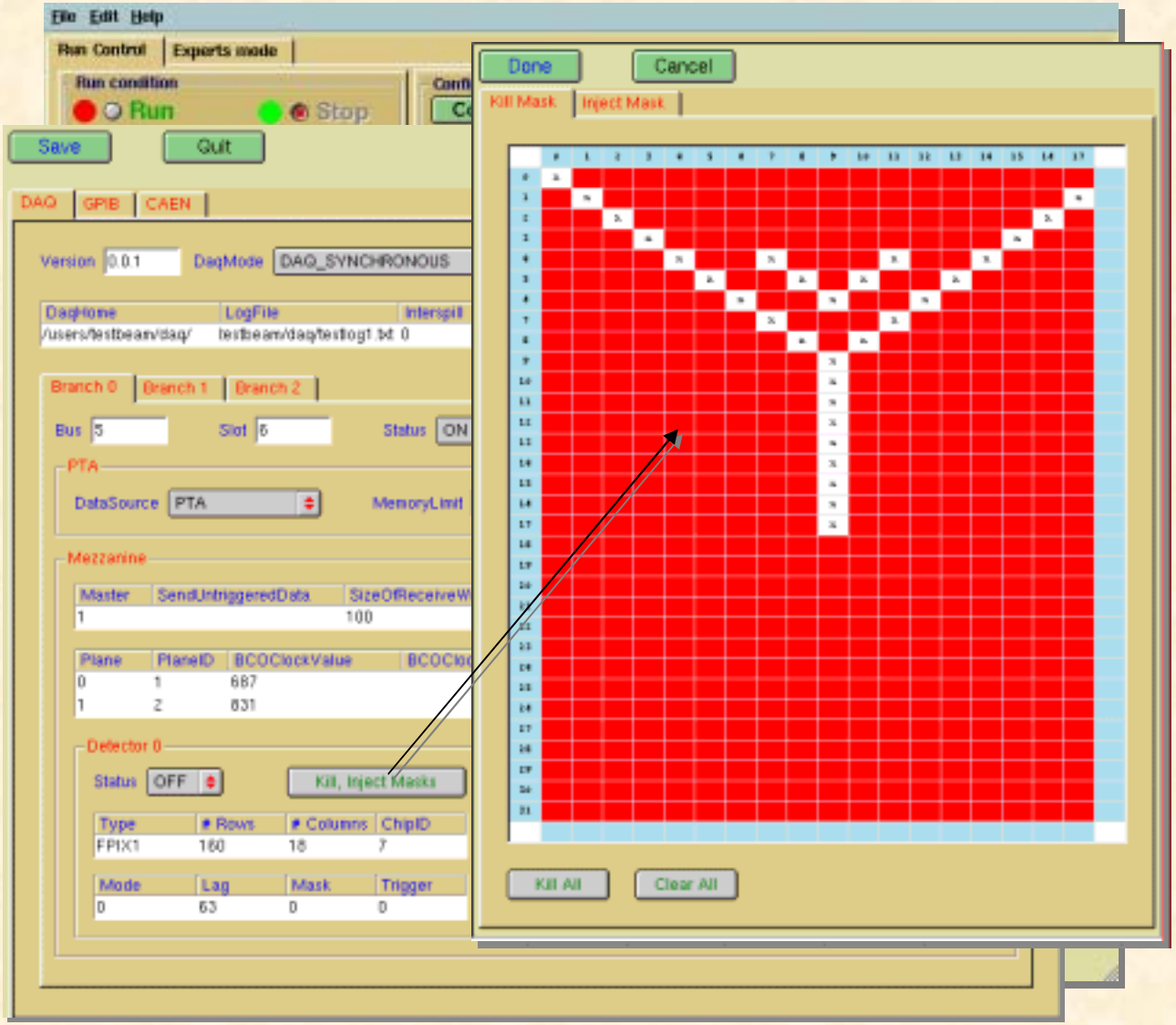
The DAQ is configured with an xml file (ASCII) that can be either edited by hand or (better) modified by a component of this GUI (Louvre)



The Pomone GUI

The DAQ is configured with an xml file (ASCII) that can be either edited by hand or (better) modified by a component of this GUI (Louvre)

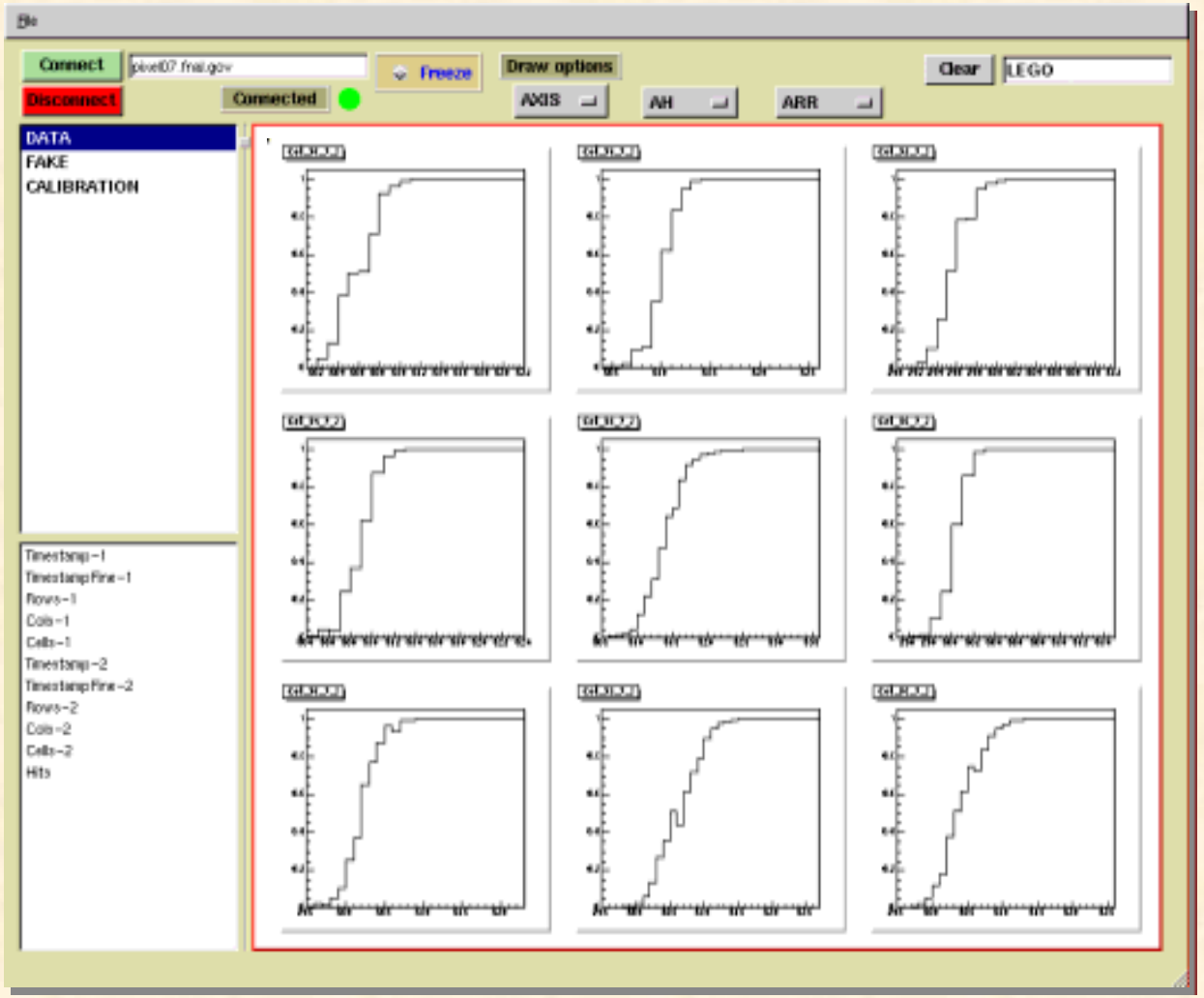
Complex patterns can be easily established to define individual pixel cells to kill in the readout or to enable/disable for charge injection



The Pomone GUI

The GUI has a button to start calibration of the pixel (see live demo).

Histograms produced by the consumer during event building are provided to users through an IP socket. Clients running everywhere can access them without placing additional burden on the data-taking CPU.



Documentation

In order to provide accurate and UP-TO-DATE documentation, we have adopted an approach we already pioneered in Focus: web based browsing of the code.

In a new approach we have bound together the three main items needed for a user to both become knowledgeable about the code and to be able to efficiently browse and navigate within:

- a reference guide (mainly intended for developers)
- a user guide (mainly intended for end-users)
- hyperlinked source code

For the documentation to be accurate, it is imperative that source code and documentation must be kept in synch AUTOMATICALLY. This can be accomplished only if the documentation consists in the comment lines of the source code itself and these accurately reflect what the code is meant to do.

We used an Open Source product to build the documentation, **Doxygen**

Check it out at http://cuf.mi.infn.it/pomone/html/index.html